

sinclair

ZX SPECTRUM

INTRODUKSJON



P.B. 73 - Økern
Oslo 5
Tlf.: 02. 64 20 65

Innhold

KAPITTEL	1	Om datamaskinen og hvordan klargjøre den til bruk. Side 5
KAPITTEL	2	Tastaturet. Side 8
KAPITTEL	3	Tall, bokstaver og datamaskinen som kalkulator. Side 11
KAPITTEL	4	Noen enkle kommandoer. Side 14
KAPITTEL	5	Enkel programmering. Side 17
KAPITTEL	6	Bruk av kassettespiller. Side 21
KAPITTEL	7	Farger. Side 25
KAPITTEL	8	Lyd. Side 27
KAPITTEL	9	Hva er inne i "boksen". Side 29

1. OM DATAMASKINEN OG HVORDAN KLARGJØRE DEN FOR BRUK.

Denne lille boken er skrevet for to typer mennesker. Den første typen er de som kan ingenting eller nesten ingenting om datamaskiner. Den andre er de som har kjennskap til datamaskin baserte systemer, men som liker å lese instruksjonsbøker før støpslet settes i veggen.

Det finnes også en tykkere bok. Dette er manualen for BASIC programmering. Nybegynnere bør ikke begynne på den manualen før introduksjonsboken er lest og forstått.

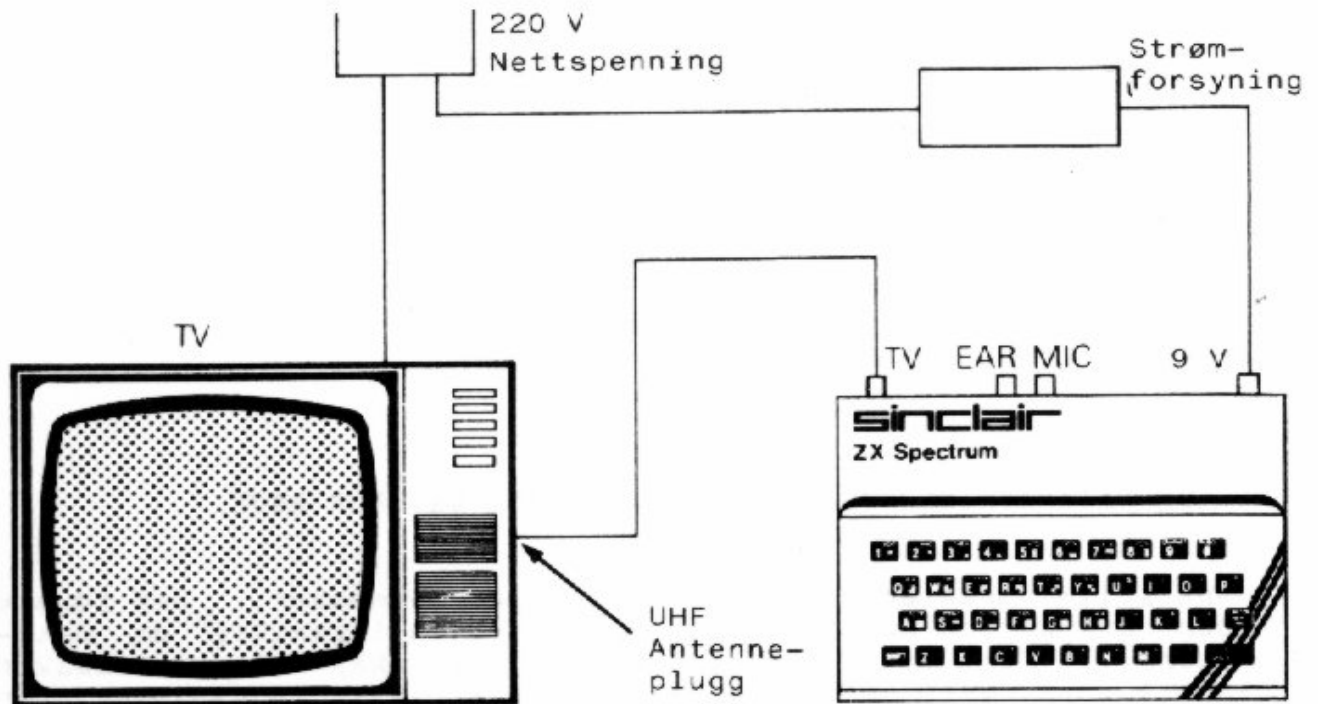
Når du pakker ut din ZX Spectrum skal du ha funnet:

1. Denne introduksjonsboken som du nå leser og manualen for BASIC programmering.
2. Datamaskinen. Den har tre kontakter jackplugger (betegnet 9V DC IN, EAR og MIC), en antenneplugg og en fremstikkende del av et kretskort hvor du kan montere ekstrautstyr. Du vil ikke finne noen bryter på maskinen - for å slå den på må du bare koble den til kraftforsyningen. For å slå den av kobler du den fra kraftforsyningen.
3. Kraftforsyningen. Denne omformer strømmen slik at ZX Spectrum kan bruke den. ZX Spectrum benytter uregulert 9 volt DC (likestrøm) 1.4 A.
4. En antenneledning som er omtrent 2 meter lang. Denne skal brukes for å forbinde datamaskinen med fjernsynsapparatet.
5. En par ledninger som er omtrent 75 cm lang med en 3,5 mm jackplugg i hver ende. Denne benyttes for å koble en båndopptaker til datamaskinen.

I tillegg til dette vil du trenge et fjernsynsapparat - ZX Spectrum fungerer uten et, men du vil ikke kunne se hva den gjør. TV-apparatet må ha UHF. Som navnet impliserer, vil ZX Spectrum lage farge signaler som på fargefjernsyn gir fargebilder. Har du bare et sort/hvit fjernsyn, vil du i stedet få bilder i sort, hvit og seks nyanser av grått. Utover dette har det ingen betydning om apparatet er sort/hvit-TV eller farge-TV.

Nå kan du koble sammen de enkelte delene som vist på tegningen som følger:

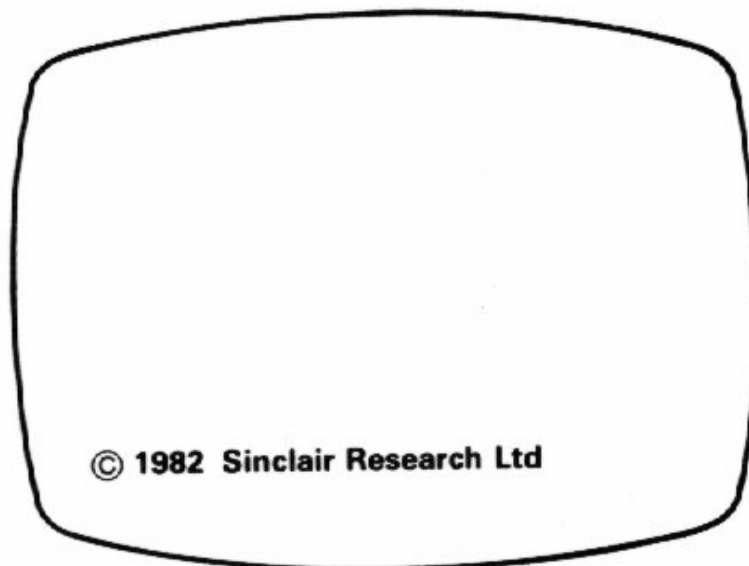
Kapittel 1



Figur 1

Hvis TV'n din har to antenneinnganger merket h.h.v. UHF og VHF, skal du benytte UHF inngangen.

Sett på strømmen til ZX Spectrum og slå på TV-apparatet. Du må nå stille TV'n inn på kanal 36 UHF. Etter å ha gjort dette skal du få følgende bilde:



Når du bruker ZX Spectrum vil du nok foretrekke at volumkontrollen settes ned til null.

Mange TV-apparater har i dag en kontinuerlig variabel kanalinnstilling. Hvis dette er tilfelle med ditt apparat,

må du bare instille denne til du får bilde som er vist i figur 2. Hvis apparatet ditt har mulighet for forhåndsinnstilling av programkanaler, er det bare å finne en ubenyttet programkanal og stille den inn på kanal 36 UHF.

Norske TV-apparater bruker et UHF-system hvor TV-bildet er bygd opp av 625 linjer og hvor bildet skifter 50 ganger i sekundet. For å overføre fargebildene benyttes et system som heter PAL. Din versjon av ZX Spectrum skal normalt kunne brukes med TV-apparater i de land som benytter de samme systemene for fjernsynsoverføringer. De fleste vest-europeiske land benyttes de samme systemene, blant unntakene er Frankrike hvor fargeoverføringssystemet heter SECAM. I U.S.A., Canada og Japan brukes helt andre systemer (i U.S.A. er f.eks. UHF standarden 525 linjer og 60 bilder pr sekund). For å bruke et TV-apparat som ikke følger norsk standard, må du ha en annen versjon av ZX Spectrum.

Når du slår av ZX Spectrum vil all informasjon (programmer og data) som er lagret i den, gå tapt. En måte å bevare denne informasjonen for ettertiden er å lagre den på kassette-bånd. Det er også mulig å kjøpe kassetter som andre har spilt inn, og på denne måten kjøre deres programmer. Den korte ledningen med en jack-plugg i hver ende kan brukes til å koble sammen en vanlig kassette-spiller med ZX Spectrum. Kapittel 8 i denne boken vil forklare dette nærmere.

Nå når du har klargjort ZX Spectrum for bruk, vil du sikkert gjerne begynne å bruke den. Resten av denne boken vil forklare hvordan du skal gjøre det. Det er nokså sannsynlig at du allerede har prøvd å trykke på tastene på tastaturet - og oppdaget at bildet med Copyright beskjedene (se figur 2) forsvant. Dette er OK, du kan ikke ødelegge datamaskinen på denne måten. Vær modig og eksperimenter. Hvis du blir stående helt fast, så kan du komme tilbake til bildet i figur 2 ved å ta ut pluggen som står i "9V DC IN" og sette den inn igjen. Dette bør være siste utvei fordi du mister all informasjon som er lagret i datamaskinen.

ADVARSEL: Prøv ikke å bruke ZX 16K RAM med ZX Spectrum. Den vil ikke fungere.

2. Tastaturet

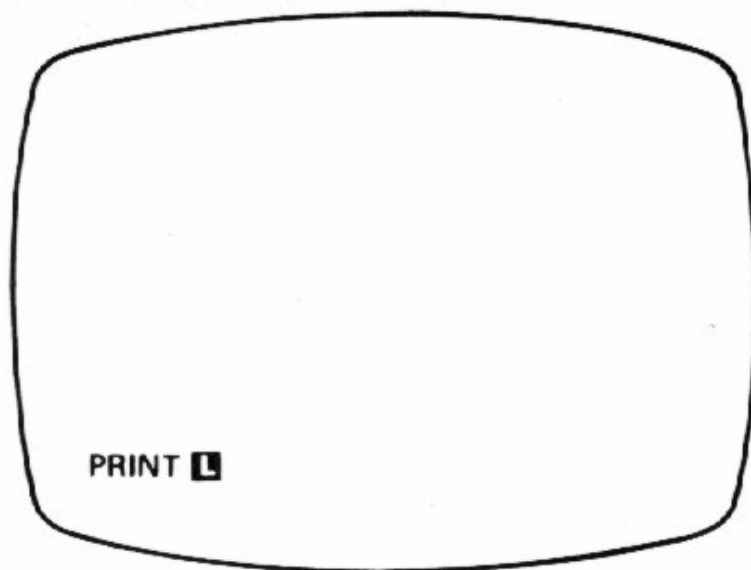
Tastaturet på ZX Spectrum er nokså likt tastaturet på en vanlig skrivemaskin. Bokstavene og tallene er på samme plass, men hver tast kan gjøre mer enn en ting. På en vanlig skrivemaskin er bokstavene vanligvis små. Trykker du på SHIFT tasten samtidig med at du trykker på en bokstav eller tall tast, vil du få store bokstaver eller tegn. Tastaturet på ZX Spectrum er også slik.

For at du skal vite hvor på skjermen den neste bokstaven eller tegnet vil bli skrevet, vil du ha en hvit-på-svart (invers video) bokstav som blinker. Blinkingen er til for at du skal kunne skille den fra eventuelle andre bokstaver på skjermen. Denne blinkende bokstaven kalles markør og vil vise forskjellige bokstaver, alt etter hva ZX Spectrum venter at du skal taste inn.

Når du slo på ZX Spectrum så fikk du et bilde med en Copyright beskjed. Hvis du så trykker på hvilken som helst bokstav, så vil ordet som står under bokstaven på den tasten komme på skjermen. Dette ordet kalles et nøkkelord. Grunnen til at du får dette på skjermen er at datamaskinen venter på en kommando fra deg for å vite hva den skal gjøre. Alle kommandoer må begynne med et nøkkelord.

Som du har sett på tastaturet finnes det tegn, symboler og ord på hver enkelt tast. For eksempel inneholder P-tasten foruten bokstaven P og ordet PRINT i hvit også tegnet " i rødt. For å få frem tegnet " på skjermen, må du trykke ned SYMBOL SHIFT samtidig som du trykker ned P-tasten. SYMBOL SHIFT finner du nær det nedre, høyre hjørnet på tastaturet.

Trykker du på P-tasten rett etter at du har slått på ZX Spectrum så vil nøkkelordet PRINT komme fram på skjermen. Bildet skulle da se ut som på figur 3:



Markøren skifter nå til ■ fordi ZX Spectrum nå venter på bokstaver eller tegn. Skriv "Hallo" - for å få frem " bruker du SYMBOLS SHIFT og " (tasten med P) samtidig. H får du fram ved å trykke på CAPS SHIFT samtidig med tasten H. (Skulle det stå noen annen tekst på skjermen før du gjør dette, så ta ut og sett inn 9V DC IN pluggen og begynn igjen.) Som en regel kan du bruke følgende: For å bruke noe som er skrevet i hvit over en tast, eller stor bokstav, må du bruke CAPS SHIFT. For å bruke et rødt tegn på en tast må du bruke SYMBOL SHIFT.

En kommando som begynner med PRINT forteller datamaskinen at den skal skrive det som står mellom anførselstegnene (") på

skjermen. For å utføre denne kommandoen må du trykke på tasten **ENTER**. Etter dette skulle skjermen vise ordet:

Hallo

og noen andre tegn. (Et blinkende spørsmålstegn betyr at det er en feil et eller annet sted. Hvis dette skjer så begynn på nytt og gjenta forsøket.) Beskjeden på bunnen av skjermen er ZX Spectrum sin måte å si at alt er 'OK' på. For øyeblikket kan du ignorere beskjeden, men når du kjører programmer er denne beskjeden viktig.

Legg også merke til en ting: Bokstaven O og tallet 0 blir skrevet ut med forskjellige symboler. Tallet null vil alltid bli skrevet med en gjennom seg (som bokstaven ø). For ZX Spectrum er et tall alltid et tall og en bokstav alltid en bokstav. Du kan derfor ikke benytte liten L for tallet 1 som du kan gjøre på enkelte skrivemaskiner.

Bokstavene E, Ø og Å finnes ikke på denne maskinen fordi den benytter det engelske alfabetet. De kan derfor ikke benyttes, men en måte å løse dette på er å bruke AE for E, OE for Ø og AA for Å, hvis du trenger disse bokstavene.

Den blinkende bokstaven **L** kalles markør. Den viser hvor på skjermen den neste bokstaven vil bli skrevet. Markøren kan vise forskjellige bokstaver. Dette gjør ZX Spectrum for å vise hvordan den vil oppfatte det neste du taster inn. Hvis markøren er **L** så betyr dette at den venter på bokstaver, tegn og tall. Er markøren derimot en blinkende **K** betyr det at den venter på et nøkkelord (på engelsk heter nøkkelord Keyword). Copyright beskjeden som du får når du slår på ZX Spectrum og alle beskjedene du får på skjermen (f.eks. alt 'OK') betyr det samme som en **K**. Et nøkkelord er et av ZX Spectrum sine spesielle ord. Datamaskinen venter at alle linjer begynner med et nøkkelord, og den sier derfor ifra at den tasten du trykker på ikke vil bli oppfattet som en bokstav - f.eks. P, men som et nøkkelord PRINT, ved å vise markøren **K**. Når den så har fått et nøkkelord, venter den ikke flere. Derfor vil den signalisere dette med å endre markøren til **L**.

De forskjellige måtene ZX Spectrum vil oppfatte et trykk på en tast på, kan kalles ofte en "tilstand". Du kjenner nå til to av disse tilstandene; **L** = "bokstav-tilstand" og **K** = "nøkkelord-tilstand". ZX Spectrum har også andre "tilstander".

Hvis du ønsker å skrive mange store bokstaver etter hverandre uten å måtte holde **CAPS SHIFT** nede hele tiden, kan du få maskinen til å skrive alle bokstaver som store bokstaver ved å trykke ned **CAPS LOCK** som du gjør ved å trykke ned **CAPS SHIFT** og 2 samtidig. (**CAPS LOCK** står skrevet med hvit skrift over 2.) For å vise at alle bokstaver heretter vil være store bokstaver, vil markøren endre seg til en **C** som viser at maskinen er i "stor bokstav tilstand". (Stor bokstav på engelsk heter Capital letter.) For å kunne skrive små bokstaver (**L** tilstand) må du taste inn **CAPS LOCK** nok en gang.

Hvis du taster **CAPS LOCK** mens ZX Spectrum venter på et nøkkelord (**K** tilstand) vil du ikke se en umiddelbar endring. Den vil fortsatt vente på dette nøkkelordet. Først når den har fått det vil du se at markøren blir **C** (for "stor bokstav tilstand") i stedet for **L** som du ellers ville få.

Kapittel 2

I tillegg til nøkkelord, bokstaver, tall og forskjellige programmerings- og matematiske uttrykk, så finnes det noen grafiske tegn. De finner du på de samme tastene som tallene 1 til 8 står på. Tegnene kan du, i likhet med tall og bokstaver, få på skjermen. For å få dette til må ZX Spectrum være i sin "grafikk-tilstand" og dette gjøres ved å taste GRAPHICS (det er CAPS SHIFT og 9 samtidig). Du vil da se at markøren endrer seg til **G**. For å komme tilbake til "bokstav tilstanden" (**L**) må du trykke ned tasten med 9 på.

Det finnes en tilstand til som maskinen kan være i. Den kan kalles "utvidet tilstand" og blir markert ved at markøren skifter til **E**. For å komme inn i denne tilstanden må du trykke ned CAPS SHIFT og SYMBOL SHIFT samtidig. Hvis du trykker ned de samme tastene ned samtidig en gang til, vil du komme tilbake til "bokstav-tilstanden".

Av og til hender det at feil taster blir trykket ned. Hittil har den eneste måten å komme videre hvis du har gjort en slik feil vært å trekke ut strøm-pluggen! Dette kan være en grei løsning når du bare har gitt ZX Spectrum en kommando. Har du derimot lagret mye informasjon i den er denne løsningen katastrofal.

Derfor finnes det en tast som heter DELETE. Den kan benyttes til å rette opp feil. For eksempel, ikke mye kan gå galt med kommandoen:

```
PRINT "Hallo"
```

La oss anta at du glemte å bruke SYMBOL SHIFT når du skulle skrive tegnet " den første gangen. Da ville skjermen vise

```
PRINT PHallo"
```

Maskinen vil ikke kjenne det som kommer etter PRINT fordi når det første anførselstegnet mangler, venter ZX Spectrum et tall - men den finner en bokstav. For å vise at den lurer på hva dette er, viser den **■** ved slutten av linjen.

Heldigvis må du ikke skrive alt omigjen. På toppen av tastaturet finner du fire piler som peker i hver sin retning og ordet DELETE. For å benytte disse må du trykke ned CAPS SHIFT samtidig som du trykker på tasten. Når du trykker ned en av pilene vil markøren bevege seg i den retningen pilen peker. DELETE tasten fungerer som et viskelær og fjerner tegnet foran markøren.

For å rette linjen med feil i så trykker du ← (CAPS SHIFT og 5 til markøren står bak P'en som er feil. Legg forresten merke til at hvis du holder tasten nede en stund vil de automatisk begynne å gjenta seg, samtidig som du hører en klikkende lyd. Alle tastene vil oppføre seg slik, ikke bare pilene. Tast så DELETE (det er CAPS SHIFT og 0) for å fjerne P'en og skriv så " (SYMBOL SHIFT og P) for å få det slik det skulle være. Legg merke til at den setter inn " uten å skrive over noe annet. Hvis du har gjort noen andre skrivefeil så kan du rette dem på samme måte; Husk bare at du ikke kan skrive over feilen, men må først fjerne feilen og så sette inn det riktige.

Hvis du nå trykker ned ENTER vil datamaskinen skrive teksten på toppen av skjermen - eller under den gamle teksten, hvis den er der fortsatt.

En komplett beskrivelse av tastaturet finnes i kapittel 1 i BASIC programmeringsmanualen.

3. Tall, bokstaver og datamaskinen som kalkulator

Du vet nå hvordan du får maskinen til å skrive bokstaver og grafikk på skjermen med PRINT. Du har også sett at ENTER må brukes for å få ZX Spectrum til å utføre den kommandoen som du skrev inn. Fra nå av vil vi i manualen ikke bruke ENTER hver gang en kommando benyttes, men forutsetter at du gjør det automatisk selv.

Datamaskinen håndterer tall lettere enn bokstaver. I det forrige kapittelet antydte vi at ZX Spectrum venter et tall etter PRINT hvis anførselstegn (") ikke brukes.

Derfor, skriver vi

```
PRINT 2
```

så vil tallet 2 komme til syne på skjermen.

Du kan også blande tall og bokstaver:

```
PRINT 2,"ABC"
```

Legg merke til at det er et mellomrom mellom 2 og ABC.

Skriv så

```
PRINT 2;"ABC"
```

og deretter

```
PRINT 2 "ABC"
```

Bruk av komma mellom det som står etter PRINT lager et mellomrom på 16 posisjoner, bruk av semi-kolon gir ikke mellomrom og ikke bruk av noe gir en feilmelding.

PRINT kan også benyttes sammen med de matematiske funksjonene som finnes på tastaturet. Dermed kan du bruke ZX Spectrum som en kalkulator.

For eksempel:

```
PRINT 2+2
```

Svaret kommer øverst på skjermen. Sammenlign dette med:

```
PRINT "2+2"
```

Kapittel 3

Det er også mulig å kombinere disse to for å få noe mer nyttig. Prøv følgende:

```
PRINT "2+2=";2+2
```

Prøv også andre regnearter:

```
PRINT 3-2
PRINT 4/5
PRINT 12#2
```

Som multiplikasjonstegn bruker ZX Spectrum tegnet * og som divisjonstegn er / benyttet.

Prøv deg med mange forskjellige regnestykker. Du kan bruke både negative tall og tall med desimaler. Husk bare at datamaskiner bruker punktum som desimaltegn og ikke komma som vi gjør i Norge.

Hvis du lager mange nok regnestykker slik at du bruker opp de 22 linjene øverst på skjermen, vil du se at noe nytt skjer. Alle linjene beveger seg en linje opp og den øverste linjen forsvinner ut av bildet. Dette kalles på engelsk "SCROLLING" og kan nærmest oversettes som "rulling".

Når du bruker ZX Spectrum som kalkulator vil du av og til merke at den ikke utfører kalkulasjonene i den rekkefølgen du ville vente deg. Prøv f.eks.:

```
PRINT 2+3#5
```

Du vil kanskje regne med at den først tar 2 og adderer 3 (= 5) og så multipliserer med 5, slik at svaret blir 25. Dette er ikke tilfelle. Multiplikasjon og divisjon utføres alltid før addisjon og subtraksjon. Derfor betyr uttrykket '2+3#5' multipliser 3 med 5, adder svaret (=15) til 2, noe som gir resultatet 17. Det er dette svaret du skulle ha på skjermen.

Fordi multiplikasjon og divisjon utføres først, sier vi at de har høyere prioritet enn addisjon og subtraksjon. I forhold til hverandre har multiplikasjon og divisjon den samme prioritet, noe som betyr at de utføres i den rekkefølge de står - fra venstre mot høyre. Når de er utført, står vi tilbake med addisjon og subtraksjon, som også har samme prioritet. De gjøres derfor også i den rekkefølge de står - fra venstre mot høyre.

La oss se hvordan ZX Spectrum ville regne:

```
PRINT 20-2#9+4/2#3
```

```
I: 20-2#9+4/2#3
II: 20-18+4/2#3
III: 20-18+2#3
IV: 20-18+6
V: 2+6
VI: 8
```

Først multiplikasjon og divisjon i rekkefølge fra venstre til høyre..

..og så addisjon og subtraksjon

Alt du trenger å vite er om en operasjon har en høyere eller lavere prioritet enn en annen. ZX Spectrum gjør dette ved å ha et tall mellom 1 og 16 til å representere prioriteten for hver operasjon.

For eksempel har * og / prioritet 8, mens + og - har prioritet 6. Denne rekkefølgen er absolutt fast, men du kan komme rundt den ved å benytte paranteser () slik at:

```
PRINT 3*2+2
```

gir svaret $6+2=8$ mens:

```
PRINT 3*(2+2)
```

gir svaret $3*4=12$.

Enkelte ganger er det nyttig å kunne gi maskinen slike uttrykk, fordi hver gang den venter ett tall fra deg kan du gi den et uttrykk (formel) som den vil finne svaret på. Unntakene fra denne regel er så få at de vil bli nevnt på en tydelig måte i hvert tilfelle.

Du kan også bruke desimaltall (husk at desimaltegnet er punktum) og vitenskaplig notasjon, noe som er meget vanlig med lommekalkulatorer. Med vitenskaplig notasjon så skriver du et vanlig tall (med eller uten desimaltegn) og en eksponent som består av bokstaven e, så eventuelt et - og deretter et tall. Eksponentdelen flytter desimaltegnet mot høyre (eller venstre hvis eksponenten er negativ), med andre ord multipliserer (eller dividerer) det opprinnelige tallet med 10 det antallet ganger eksponenten sier. For eksempel:

```
2.34e0=2.34
2.34e3=2340
2.34e-2=0.0234
```

Forsøk å skrive dette på datamaskinen. Dette er en av de få tilfellene hvor du ikke kan erstatte et tall med et uttrykk (formel); du kan f. eks. ikke skrive:

```
(1.34+1)e(6/2)
```

Du kan også benytte uttrykk (formler) hvis verdi ikke er tall, men en rekke (streng) med bokstaver. Du har gjentatte ganger sett den enkleste formen for dette; en streng med bokstaver skrevet med anførselstegn foran og bak ("ABC"). Dette ligner den enkleste form for tall uttrykk, som er tall skrevet ned alene. Det du ikke har prøvd er bruken av + i sammenheng med strenger av bokstaver (du kan ikke benytte -, * eller /, så det er ikke problemer med prioriteten i dette tilfelle). Du kan sette sammen strenger med bokstaver, så prøv:

```
PRINT "Sincla"+"ir ZX Spec"+"trun"
```

Du kan legge sammen så mange strenger du vil i et eneste uttrykk og du kan også, om du vil, benytte paranteser.

4. Noen enkle kommandoer

Datamaskinens hukommelse kan benyttes til å lagre mange forskjellige ting. Hittil har du sett at PRINT kommandoen tillater deg å vise tall, bokstaver og resultatet av regnestykker som benytter både tall og bokstaver, på skjermen.

Hvis vi ønsker å fortelle datamaskinen at den skal huske et tall eller en streng med bokstaver, må vi sette av plass i hukommelsen for dette.

De fleste lommekalkulatorene har en tast som heter 'minne' som benyttes for å huske tall for senere bruk. Din ZX Spectrum kan gjøre mer enn det. Den kan ha så mange lagringsplasser som du vil, og du kan sette navn på hver eneste lagringsplass.

Som et eksempel; anta at du ønsker at den skal huske din alder (som vi sier er 34 år). For å få dette til, må du benytte LET kommandoen (LET er nøkkelordet på L-tasten).

```
LETalder=34
```

Når du bruker LET kommandoen setter du av et område i datamaskinen, kaller det 'alder' og lagrer tallet 34 i dette området. For å få tilbake denne informasjonen må du skrive:

```
PRINT alder
```

og tallet 34 skal komme fram på skjermen. Tallet 34 er fortsatt lagret i maskinens hukommelse. Hvis 34 er feil alder, du er egentlig 36, kan du endre det tallet som er lagret ved å skrive:

```
LET alder=36
```

skriv så:

```
PRINT alder
```

og 36 skulle komme fram på skjermen. Vi kaller 'alder' en variabel fordi innholdet i den kan variere. Du kan skrive en beskjed sammen med verdien av en variabel. Skriv:

```
PRINT "Din alder er "; alder
```

Datamaskinen kan huske mer enn tall med et navn ('alder') knyttet til det. Den kan også huske strenger med bokstaver. For å skille mellom tall variable og streng variable - som de kalles - så bruker vi tegnet \$ etter navnet på streng variabelen.

For eksempel, ønsker du å lagre denne strengen med bokstaver:

```
"Din alder er"
```


kan du kalle den

a\$

(streng variable kan bare ha et navn med en bokstav foruten \$).
Skriv så:

```
LET a$="Din alder er "
```

Hvis du så skriver

```
PRINT a$
```

vil strengen med bokstaver komme tilbake på skjermen.

Hvis du ikke har slått datamaskinen av etter at du begynte på dette kapittelet, kan du skrive

```
PRINT a$;alder
```

og se hva som skjer.

Det finnes også andre måter for å få informasjon inn i datamaskinens hukommelse (dvs. uten å bruke LET).

Et eksempel er INPUT kommandoen som i sin enkleste form forteller datamaskinen at informasjon er å vente fra tastaturet. I stedet for å skrive LET hver eneste gang, kan du skrive

```
INPUT alder
```

Med en gang du så har trykket ned ENTER tasten, vil en blinkende **L** tre fram på skjermen. Dette betyr at datamaskinen ønsker informasjon fra deg. Det du gjør er å taste inn din alder (f.eks. 56) og så trykke på ENTER tasten. Selv om ingenting ser ut til å skje, har variabelen 'alder' nå fått den nye verdien du tastet inn. Skriver du så

```
PRINT alder
```

skulle bevise dette.

La oss så kombinere alt dette i en rekke med kommandoer. Skriv:

```
LET b$="Hva er alderen din?"
LET a$="Din alder er "
INPUT (b$);alder: PRINT a$;alder
```

Legg merke til at den siste linjen består av to kommandoer, adskilt med kolon.

Kapittel 4

```
INPUT (b$);alder
```

er en annen måte å skrive

```
INPUT "Hva er alderen din?"; alder
```

5. Enkel programmering

Til nå har vi fortalt datamaskinen hva den skal gjøre direkte fra tastaturet. Selv om det er mulig å utføre mer enn en kommando på denne måten, har den dog begrensede anvendelsesmuligheter.

Hovedpoenget med datamaskiner er at de programmerbare. Det betyr at vi kan gi dem en serie med instruksjoner og få dem til å gjøre ting i rekkefølge.

Hver datamaskin har et eget språk som gjør at vi kan kommunisere med den. Enkelte språk er enkle, slik at datamaskinen lett kan forstå dem. Uheldigvis er slike språk vanskelige å forstå for mennesker. Det motsatte er også tilfelle - språk vi lett kan forstå er vanskelige for maskinen, og må derfor oversettes eller tolkes.

ZX Spectrum benytter et høynivå språk som heter BASIC.

BASIC står for Beginners All-purpose Symbolic Instruction Code, og språket ble laget ved Dartmouth College i New Hampshire, USA, i 1964. Det er i omfattende bruk på personlige datamaskiner. Uheldigvis, selv om BASIC i grove trekk er lik for alle versjoner, så er det små og viktige forskjeller mellom de enkelte 'dialektene'. Denne håndboken er skrevet spesielt for ZX Spectrum. ZX Spectrum BASIC er ikke så forskjellig fra en (ikke eksisterende) felles BASIC, så det skulle ikke være så altfor store problemer med å tilpasse programmer skrevet i andre dialekter slik at de kan kjøres på ZX Spectrum. En ting er viktig; Til forskjell fra andre BASIC versjoner tillater ikke ZX Spectrum BASIC at nøkkelordet LET utelates når du gir verdier til variable.

Det er en grense for hvor mange instruksjoner som kan lagres i datamaskinen. ZX Spectrum sier ifra når den grensen nås med en 'buzz'-lyd.

Når du programmerer i BASIC er det nødvendig å si i hvilken rekkefølge instruksjonen skal utføres. Derfor har hver linje med instruksjoner et nummer på begynnelsen. Det er vanlig å begynne med 10 og øke tallet med 10 for hver ny linje. Dette gjør at ny linjer kan settes inn mellom to eksisterende linjer, hvis programmet må modifiseres.

La oss se på et enkelt program. Gå tilbake til kommandoene i slutten av forrige kapittel. Hvis vi ønsker å gjenta denne serien med kommandoer er det nødvendig å taste de inn for hver gang. Et program gjør dette overflødig.

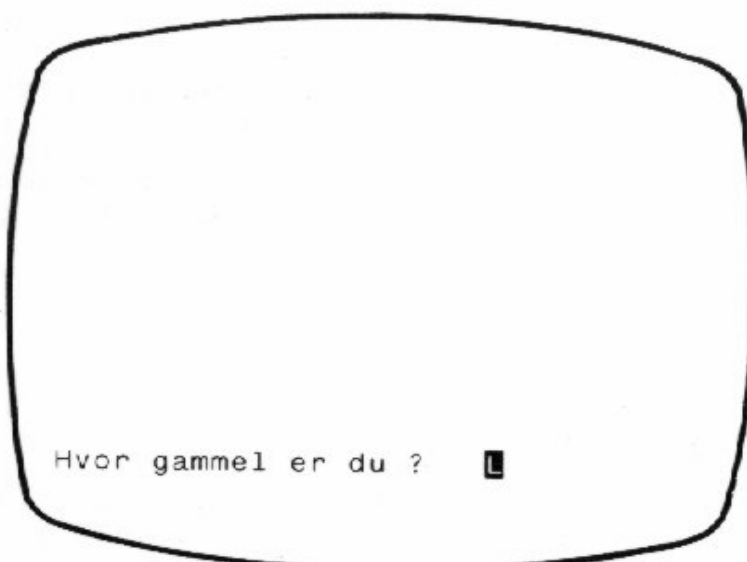
Skriv dette - husk å taste ENTER etter hver linje.

```
10 LET b$="Hvor gammel er du? "  
20 LET a$="Din alder er "  
30 INPUT (b$); alder  
40 PRINT a$; alder
```

Legg merke til at det ikke er nødvendig å taste inn noen mellomrom - untatt innenfor anførselstegnene.

Ingenting vil skje før vi forteller datamaskinen at den skal begynne å arbeide med programmet. Det gjøres ved å bruke RUN (nøkkelordet på R-tasten).

Tast inn denne kommandoen og se hva som skjer.



Hvor gammel er du ? █

Du har kanskje lagt merke til en pil som peker mot høyre hver gang du har tastet inn en ny linje. Denne peker på den sist inntastede linjen. Hvis du ønsker å se programmet en gang til, skriv **ENTER** (eller **LIST**). Du kan også benytte **RUN** for å kjøre programmet så mange ganger du selv ønsker. Hvis du ikke trenger dette programmet mer, kan du slette det fra minnet ved å bruke **NEW** kommandoen. Denne sletter ut programmet og gir deg et blankt 'ark' som du kan skrive inn et nytt på.

Skriv **NEW** og så **LIST** og se hva som skjer.



© 1982 Sinclair Research Ltd

For å repetere:

Når du skriver inn en kommando med et nummer foran, så forteller du datamaskinen at dette ikke er en enkel kommando, men en programlinje. Datamaskinen utfører ikke instruksjonen med en gang, men lagrer den for senere bruk.

ZX Spectrum skriver på skjermen (eller lister) alle linjene som du har skrevet inn med en **↵** mot den siste linjen du skrev.

Datamaskinen utfører ikke kommandoen i disse linjene med en gang, men lagrer de.

For å få maskinen til å utføre kommandoene må du bruke **RUN**-kommandoen.

Taster du inn **ENTER** vil du få listen med linjer tilbake på skjermen.

Prøv et nytt enkelt program. Dette skal være litt mer matematisk, og skrive ut kvadratet av alle hele tall mellom 1 og 10 (kvadratet av et tall er det samme som tallet multiplisert med seg selv en gang).

For å få tallene mellom 1 og 10 må du benytte et annet konsept i BASIC-programmering. Det er metoden som vi bruker for å få datamaskinen til å telle. Tidligere har du sett at tall kan lagres i hukommelsen med et 'navn' tilknyttet seg - eller rent formelt: Å gi variabelen en verdi. Du kan la variabelen **x** begynne med verdien 1 og øke verdien i trinn på 1 til verdien er 10. Til dette benytter du **FOR...NEXT...STEP** kommandoen.

For å kunne skrive inn dette programmet må du først bruke **NEW** for å bli kvitt det gamle. Skriv så:

```
10 FOR x=1 TO 10 STEP 1
```

(Vanligvis kan du sløyfe **STEP 1** hvis du benytter 1 som trinn).

Den neste linjen må fortelle datamaskinen hva den skal gjøre med **x** uansett hvilken verdi den har. Skriv:

```
20 PRINT x,x*x
```

Til slutt trenger vi en linje som forteller maskinen at den skal gå til neste verdi av **x** - skriv derfor:

```
30 NEXT x
```

Når maskinen kommer til denne kommandoen vil den gå tilbake til linje 10 og gjenta instruksjonene. Når så **x** overstiger 10 vil den fortsette til neste linje i programmet dvs linje 40.

Programmet skulle nå se ut som følger på skjermen:

```
10 FOR x=1 TO 10 STEP 1
20 PRINT x,x*x
30 NEXT x
```

For å gjøre programmet komplett må vi fortelle maskinen at programmet er slutt når **x=10**, så skriv:

```
40 STOP
```

Kapittel 5

Hvis du nå kjører programmet (kommandoen RUN starter programmet) skal du få to tall kolonner. Den første viser verdien av x , den andre verdien $x*x$ eller kvadratet av x . Du kan sette navn på disse kolonnene ved å skrive inn en ny programlinje:

```
5 PRINT "x", "x*x"
```

Legg merke til at denne linjen har et lavere linjenummer enn de andre linjene, selv om den ble skrevet inn etter de. Dette linjenummeret gjør at linjen kommer på riktig plass i programmet.

Prøv å skrive andre programmer hvor du bruker andre matematiske instruksjoner. Hvis du er i tvil om hvordan de skal benyttes kan du se i BASIC programmeringsmanualen hvor de er omtalt.

6. Bruk av kassettspilleren

Det er tungvint å måtte skrive inn programmene hver gang du ønsker å kjøre dem. Derfor har ZX Spectrum muligheten for å lagre programmene på magnetbånd. Dette kan gjøres med en vanlig kassettspiller. Hvis du har et program i minnet kan du forsøke å lagre det med følgende prosedyre:

Hvis du kan lagre programmer på magnetbånd kan du også lese de tilbake (spille av) senere.

De fleste kassettspillere kan benyttes for datamaskinens vedkommende. De billige portable modellene er minst like gode som de dyre stereo modellene - og gir mindre problemer i tillegg. Det er en fordel om spilleren har et telleverk for båndet.

Opptakeren må ha en kontakt for tilkobling av mikrofon og en kontakt for øretelefoner (hvis den ikke har det, prøv kontakten for utvendige høyttalere). Kontaktene må være for 3,5 mm jack-plugger slik at ledningene som fulgte med kan benyttes. Andre ledningstyper kan gi et for svakt signal for datamaskinen.

Hvilke som helst kassetttyper kan benyttes, selv om 'LOW NOISE' typen er å foretrekke.

Du kobler kassettspilleren til ZX Spectrum med de to ledningene som fulgte med. Den ene ledningen kobler du mellom mikrofoninngangen på spilleren og MIC-kontakten på baksiden av ZX Spectrum. Den andre kobles mellom øretelefon uttaket på spilleren og EAR-kontakten på datamaskinen. Det er ikke mulig å skade ZX Spectrum ved å feilkoble ledningene.

Når du bruker **SAVE** kommandoen for å lagre et program på magnetbånd, må du forsikre deg om at en av pluggene på ledningen som forbinder 'EAR' kontaktene er trukket ut - det har ingen betydning hvilken av dem som trekkes ut. Glemmer du å gjøre det vil det bli spilt inn en jevn tone på båndet, og programmet lagres ikke. Grunnen til dette er at når spilleren tar opp, forsterker den signalet som kommer inn gjennom MIC kontakten og sender det ut gjennom EAR kontakten. Hvis signalet kommer tilbake til datamaskinen blir det en sirkel hvor signalet vil slette seg selv. Dermed vil du ikke få lagret det du ønsker.

Skriv et eller annet program inn i maskinen, f.eks. kvadratprogrammet fra forrige kapittel, og skriv så:

SAVE "Kvadrat"

Kvadrat er bare et navn som du setter på programmet når det er på båndet. Du kan bruke opptil 10 tegn i navnet, som alle må være enter bokstaver eller tall.

ZX Spectrum kommer så med en beskjed på skjermen: **Start tape then press any key.** Dette betyr: "Start båndet og trykk inn hvilken som helst tast".

La oss først ta en generalprøve uten at du starter båndet, slik at du kan se hva som skjer. Trykk derfor ned en tast uten at du har startet båndet. Legg merke til kanten rundt TV-skjermen - der vil du se et mønster av fargede striper.

Fem sekunder med røde og (bleke) blå striper, omtrent en centimeter brede, som beveger seg sakte oppover.

Et kort øyeblikk med blå og gule striper.

Et (1) sekund hvor alt er som normalt.

Kapittel 6

To sekunder med røde og (blek) blå striper igjen.

Omtrent 1 sekund med de blå og gule stripene.

Prøv omigjen til du greier å skille alle disse hendingene. Informasjonen som er lagret i minne lagres på båndet i to puljer (blokker). Begge blokkene har en innledning, som er det røde og (blek) blå mønsteret, og informasjonen, som er det blå og gule mønsteret. Den første blokken er en foreløpig blokk som inneholder navnet og en del annen informasjon om programmet, den andre er programmet selv med de variablene som måtte være tilstede. Det hvite mellomrommet er...et mellomrom.

Så prøver vi å lagre signalet på båndet:

1. Spol båndet fram til et område hvor det ikke er tatt opp noe, eller som du vill spille inn over det som er der.

2. Skriv:

SAVE "Kvadrat" (og ENTER)

3. Kassettspilleren startes på opptak.

4. Trykk ned hvilken som helst tast på ZX Spectrum.

5. Se på TV-skjermen som tidligere. Når ZX Spectrum er ferdig (med beskjeden 0 OK på skjermen) stopper du kassetteopptakeren.

For å være sikker på at alt har gått som det skulle kan du kontrollere det som er lagret på båndet med det som er i datamaskinen ved å bruke VERIFY kommandoen. Du gjør som følger:

1. Skru lydstyrken (volumet) opp omtren halvveis av det som er mulig. Kobl til 'EAR'-ledningen.

2. Spol kassetten tilbake til et punkt før du begynte innspillingen.

3. Skriv:

VERIFY "Kvadrat"

(VERIFY er tilgjengelig i 'extended mode', dvs. i 'utvidet tilstand, deretter 'shift' R).

4. Sett igang kassettspilleren på avspilling.

Mønsteret langs borden rundt skjermen vil skifte mellom rødt og (blek) blått helt til båndet er der hvor du gjorde opptaket. Derfa vil du se det samme som da du spilte inn programmet. I det mellomrommet på midten av innspillingen, som varte ett sekund, vil Program Kvadrat bli skrevet på skjermen - når maskinen leter etter noe vil den skrive navnet på alt den kommer over. Hvis du ser alt dette og maskinen stopper med beskjeden 0 OK, er programmet trygt lagret og du kan hoppe over de neste linjene.

Hvordan sikre seg at programmet er lagret.

Har navnet kommet opp på skjermen?

Hvis ikke, så ble programmet enten ikke lagret (spilt inn) skikkelig, eller lesingen (avspillingen) var ufullstendig. Du må finne ut hva som er tilfelle. For å se om innspillingen var OK, spol båndet tilbake til der du begynte innspillingen. Spill så av båndet gjennom båndopptakerens egen høttaler - du må sannsynligvis koble fra ledningen på høretelefonkontakten på opptakeren. Den røde og (blek) blå innledningen gir en konstant, høy tone. Den blå og gule delen gir en mer ubehagelig lyd, omtrent som en morse beskjed i en orkan. Begge disse lyd-mønstrene er kraftige - med full lydstyrke kan de lett overdøve en samtale.

Hører du ikke disse lydene er programmet neppe lagret. Kontroller at ledningene forbinder de riktige kontaktene. Forsikr deg om at MIC kontaktene er forbundet med hverandre og at EAR kontaktene ikke er det. På noen båndspillere hender det at det ikke er kontakt hvis jack-pluggen er skjøvet helt inn. Prøv derfor å trekke den ørlite grann (1/4 mm) ut - av og til kan du kjenne at den finner et naturlig 'hvilested'. Kontroller videre at du ikke forsøkte å spille inn på den klare plastikken på begynnelsen av båndet - den har ikke noe magnetbelegg. Når du har kontrollert dette - prøv å lagre på nytt.

Hvis du kan høre de beskrevne lydene så er troligvis **SAVE** kommandoen utført korrekt. Problemet er da innlesingen (avspillingen) av programmet.

Kontroller ledningene igjen. Deretter sjekker du lydstyrken (volumet). Hvis lyden er for svak vil ikke datamaskinen høre signalet skikkelig, og du vil ikke se de riktige mønstrene på TV-skjermen. Er lydsignalet for sterkt vil det bli forvrengt (du kan kanskje høre det gjennom datamaskinens egen høyttaler). Mellom disse to ytterpunktene er det mange akseptable instillinger, men du kan jo forsøke deg fram.

Det neste tilfellet intreffer når maskinen finner programmet og skriver navnet på skjermen. Hvis feilen er av denne typen er det følgende muligheter:

Du tastet inn feil navn, enten i **SAVE** (her skriver maskinen feil navn på skjermen) eller i **VERIFY** kommandoen - maskinen ingorerer programmet og viser rødt og blått mens den er på sin vei.

Det er en feil på båndet. I dette tilfellet gir den beskjeden **R Tape loading error**, noe som betyr at den ikke kunne bekrefte at programmet var korrekt lagret. Du må lagre det på nytt.

Det er mulig at lydstyrkeinstillingen (volumet) ikke er helt riktig. Det kan alikevel ikke være så altfor galt, for datamaskinen greide å lese den første blokken.

Kapittel 6

La oss gå utifra at du ha fått lagret programmet og fått bekreftet riktig lagring. Å lese det tilbake i maskinen er omtrent som å bekrefte riktig lagring, du bare skriver:

LOAD "Kvadrat"

i stedet for

VERIFY "Kvadrat"

LOAD finner du på J-tasten. Når du fikk bekreftet korrekt innspilling skulle du ikke få noen problemer når du leser programmet tilbake.

LOAD fjerner det gamle programmet og alle variable som er i datamaskinens hukommelse før det nye programmet leses inn fra båndet.

Når du har lest inn et program kan du kjøre det med kommandoen **RUN**.

Det er mulig å kjøpe ferdiginnspilte programmer på kassette. De må være skrevet for ZX Spectrum; andre typer datamaskiner benytter andre innspillingsteknikker slik at det ikke går å bruke kassetten på andre maskiner.

Hvis du har lagret mer enn ett program på samme side, så vil de alle ha et navn. Du velger hvilket du vil lese inn i **LOAD** kommandoen. Ønsker du å lese inn 'helikopter' skriver du:

LOAD "helikopter"

(**LOAD ""** betyr les inn det første programmet du kommer over. Denne kommandoen er meget effektiv hvis du ikke husker navnet på programmet ditt.)

7. Farger

En av grunnene for å kjøpe ZX Spectrum er dens evne til å bruke farger på TV-skjermen. Skjermen er delt i to områder. Den ytre delen kalles **BORDER** og den sentrale delen **PAPER**. Du kan skifte farge på disse delene som du selv måtte ønske, både fra tastaturet og i et program.

ZX Spectrum har 8 farger som kan benyttes. De har et nummer mellom 0 og 7. Selv om det virker som om fargene har fått et vilkårlig nummer, gir de en avtagende grå-sjattering på sort/hvit fjernsyn.

Her følger en liste - du finner fargene igjen over de samme tallene på tastaturet.

0	Sort
1	Blå
2	Rød
3	Fiolett
4	Grønn
5	Cyan (blek blå)
6	Gul
7	Hvit

Når maskinen slås på arbeider den i sort - hvitt. Med andre ord er normal verdi for **BORDER** og **PAPER** 7 - hvitt. Bokstavfargen på skjermen bestemmes av kommandoen **INK**. Vanligvis er denne 0, dvs. sort. Nå du starter bestemmer ZX Spectrum fargene selv. Du kan imidlertid endre fargene. Skriv f.eks.:

BORDER 2

Hvis du husket å taste **ENTER** til slutt, skulle den ytre delen (**BORDER**) skifte fra hvit til rød. Dette inkluderer det området hvor kommandoer og instruksjoner tastes inn. Prøv selv med andre farger.

Neste skritt er å skifte farge på det sentrale området ved å skrive:

PAPER 5

PAPER er en av 'extended mode' (utvidet tilstand) kommandoene som tidligere nevnt. Den kommer du i ved å trykke på **CAPS SHIFT** og **SYMBOLS SHIFT** samtidig. **PAPER** er da 'shift' C. Når du så trykker to ganger på **ENTER** skulle det midtre området på skjermen skifte farge til Cyan (blek blått). Den første **ENTER** kommando opphever den **PAPER** som allerede er i datamaskinen. Først når den andre **ENTER** kommando tastes inn (og får maskinen til å **LISTE** programmet som er i minnet og dermed gjenoppbygge informasjonen om skjermminnholdet) brukes **PAPER** kommando og den får den nye fargen. Hvis du bruker et farge-TV og fargene ikke skifter, prøv med å justere fargeinstillingen (og kanskje kanalinstillingen).

Kapittel 7

INK kommandoen er lik **PAPER** kommandoen og kontrollerer fargen på skriften som kommer på **PAPER** området på skjermen. Hvis du gir de begge samme fargen, vil naturligvis ikke bli noe synlig skrift på skjermen.

Kommandoene **BORDER**, **PAPER**, og **INK** kan brukes i programmer. Her følger et enkelt et som viser de fargemulighetene som finnes:

```
10 FOR x=0 TO 7
20 BORDER x
30 PAPER 7-x: CLS
40 PAUSE 50
NEXT x
```

Dette programmet vil gå igjennom alle 8 fargene med forskjellig farge på **BORDER** og **PAPER** hver gang. **CLS** kommandoen etter **PAPER** tvinger datamaskinen til å gjenoppbygge skjermbildet med den nye **PAPER** fargen. **PAUSE** kommandoen sørger for at kjøringen av programmet stopper 1 sekund så det er mulig å se hva som skjer (du kan jo kjøre programmet uten denne kommandoen også for å se hva som skjer). For å vise hvordan **INK** kommandoen fungerer kan du prøve dette programmet, etter at du har gitt **NEW** kommandoen:

```
10 BORDER 7
20 PAPER 1
30 INK 4
40 PRINT "Grønne bokstaver på blå bakgrunn"
```

Det finnes også andre kommandoer som benytter fargemulighetene som **ZX Spectrum** har. Disse er forklart i **BASIC** programmeringsmanualen.

8. Lyd

ZX Spectrum kan lage lyder med nesten ubegrenset variasjon. Tonefrekvensen og varigheten kontrolleres av deg. Kommandoen BEEP eksisterer for at du skal kunne få maskinen til å gi lyd fra seg. BEEP er en 'extended mode' (utvidet tilstand) kommando som du får ved bruk av Z-tasten.

Midtfrekvensen for BEEP kommandoen er den midtre C. Denne frekvensen kan varieres innenfor BEEP kommandoen og hvilken som helst tone kan lages hvis den kan uttrykkes som semitoner eller deler av semitoner over eller under dene sentral-frekvensen. Hvis du skriver kommandoen:

BEEP 2,0

(husk ENTER) så skulle datamaskinen lage tonen midtre C i to sekunder.

De to tallen kontrollerer i fellesskap den lyden som kommer ut - den første: lengden av tonen i sekunder; den andre: tonehøyden i semitoner over/under midtre C. Verdien for midtre C er 0, C♯ er 1, D er 2 osv. oppover til neste C som er 12 (en oktav består av 12 semitoner). Du kan gå videre og bruke 13, 14...osv., slik at jo høyere tall, desto høyere frekvens.

Prøv dette:

BEEP 1,4: BEEP 1,2: BEEP 2,0

Du skal nå kunne høre den første takten av 'Three Blind Mice'. Fordi det er mulig å koble ganske mange BEEP på denne måten (:), så kan du, hvis du er tolmodig, lage en hel melodi.

(Kolon kan binde sammen en eller flere av de grunnleggende kommandoene.)

Her er et mere sammensatt eksempel; du kan lage en syngende kameleon med å blande BEEP og BORDER sammen:

**BORDER 1: BEEP 1,14: BORDER 3: BEEP 1,16:
BORDER 4: BEEP 1,12: BORDER 6: BEEP 1,0:
BORDER 5: BEEP 4,7: BORDER 1**

Du behøver ikke bekymre deg over at dette strekker seg over mer enn en linje. Datamaskinen bryr seg ikke om det.

Et kort program som spiller en hel serie med noter kan være som følger:

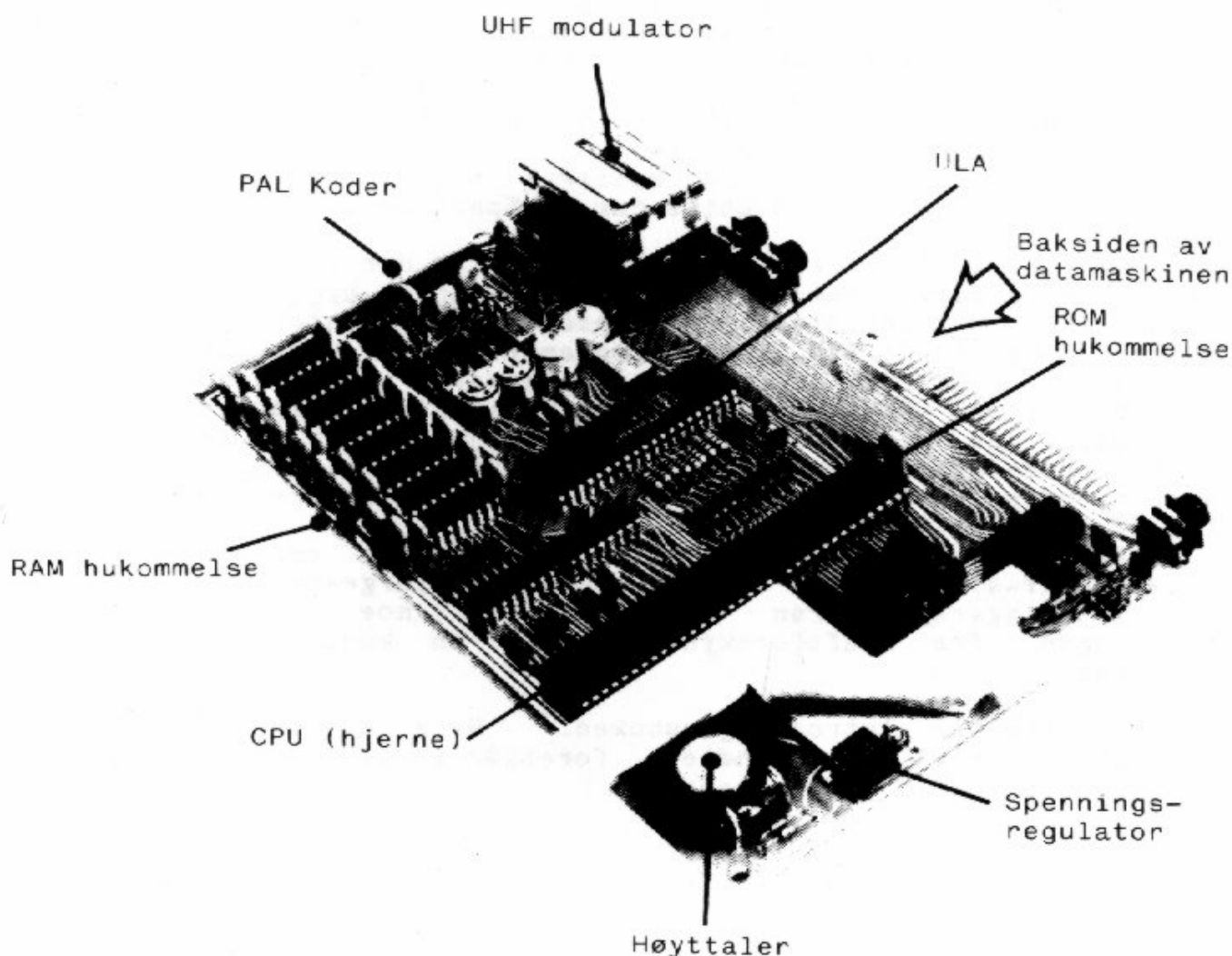
```
10 FOR x=0 TO 24
20 BEEP 2,x
30 NEXT x
```

Du kan gjøre mye mer med denne kommandoen - se i manualen for BASIC programmering for andre ideer.

For toner lavere en den midtre C indikerer du antallet semitoner med et negativt tall.

9. Hva er inne i boksen

Bildet under viser hvordan ZX Spectrum ser ut inni.



Som du kan se har alle delene en forkortelse på tre bokstaver. De sorte, rektangulære plastikk stykkene med mange metallben er integrerte kretser som faktisk gjør alt arbeide. Inne i hver av disse er det et silikon stykke (1/4" x 1/4") som er koblet til benene med ledninger. På disse silikon stykkene finnes det tusenvis av transistorer som til sammen utgjør de elektroniske kretsene som er datamaskinen.

Hjernen bak det hele er prosessoren, ofte kalt CPU. I ZX Spectrum er det en Z80A, som er en raskere versjon av den populære Z80.

Prosessoren kontrollerer datamaskinen, utfører regnearbeidet, finner ut hvilke taster som er trykket på, beslutter hva som skal gjøres pga trykningen og generelt bestemmer hva datamaskinen skal gjøre. Dog, selv om den er smart, kan den ikke gjøre dette alene. Den kan ingenting om

Kapittel 2

BASIC eller desimalregning f.eks., og den må få alle sine instruksjoner fra en annen krets, ROM som står for 'Read Only Memory' eller 'Bare Lesbart Minne'. Denne kretsen inneholder en lang rekke instruksjoner som utgjør et datamaskinprogram som kan fortelle prosessoren hva den skal gjøre i alle mulige situasjoner. Programmet er ikke skrevet i BASIC, men i Z80 maskin kode. Dette er en lang rekke tall som representerer instruksjonene. Totalt er det 16384 (16*1024) slike instruksjoner, og dette er grunnen til at ZX Spectrum BASIC av og til kalles 16K BASIC (1K er 1024)

Selv om det er liknende kretser i andre datamaskiner, er alle disse instruksjonene unike for ZX Spectrum og er skrevet for den.

De neste 8 byggeklossene ved siden av den er minnet eller RAM (Random Access Memory) og det er to andre byggeklosser som samarbeider nært med disse. Prosessoren lagrer informasjonen den ønsker å beholde i RAM, alle programmer, variable, bildet som skal vises på TV'n og andre ting den ønsker å holde rede på.

Den store byggeklossen er SCL (Sinclair Computer Logic) kretsen. Den opptrer som et kommunikasjonscenter og sørger for at alt prosessoren krever blir gjort. Den leser også minnet og ser hvordan TV bildet skal være, og sender tilsvarende signaler til TV-tilknytningen.

PAL-koderen er en samling komponenter som omvandler SCL-kretsens TV-signaler til en form som passer farge-tv apparatene.

Spenningsregulatoren omformer den noe uregelmessige spenningen fra kraftforskyningen til en konstant 5 volts spenning.

Dette avslutter Introduksjonsboken. Hvis du synes du har forstått dette vi har behandlet, foreslår vi at du prøver deg på manualen for BASIC programmering.



P.B. 73 - Økern
Oslo 5
Tlf.: 02. 64 20 65